

SINGLE SIGN-ON FOR ACCESS TO A CENTRAL DATA REPOSITORY**Related Applications**

[001] The present application claims the benefit of U.S. Provisional Patent Application Serial No. 60/238,847 filed October 6, 2000, and is further a continuation-in-part of U.S. Application Serial No. 09/933,567 filed August 20, 2001, which claims the benefit of U.S. Provisional Patent Application Serial No. 60/226,117, filed August 18, 2000, which is further a continuation-in-part of U.S. Application Ser. No. 09/923,285 filed on August 6, 2001, which claims the benefit of U.S. Provisional Patent Application Serial No. 60/223,232, filed August 4, 2000, all of which are hereby incorporated by reference as if set forth fully herein.

Technical Field

[002] The field of the present invention relates generally to systems and methods for the storage, management, and delivery of user or consumer information on or over a network. More particularly, the present invention relates to systems and methods for providing access to user or consumer information from various endpoints on or over a network.

Background of the Invention

[003] As information technology and network technology become more prolific, people find themselves repeatedly and manually inputting the same data into different computer systems. For example, consumers may find themselves having to manually input their personal and billing information via each vendor website through which they choose to complete an electronic commerce ("e-commerce") or mobile commerce ("m-commerce") transaction. As the number of secure websites grows,

consumers also find themselves having to manage numerous usernames and passwords. Thus, there is a need for a convenient and secure system for automating the management of consumer information.

[004] Automated or partially automated solutions for managing information historically have largely been localized processes. Using conventional techniques, users are able to create and store data files containing personal information on their personal computers or other client devices, such as personal digital assistants ("PDAs"), pagers, mobile telephones, etc. The data elements in such data files can be shared using specialized applications for filtering data out of the data file and into another application. However, such systems typically require a permanent download of proprietary data management software that might not be compatible among different devices. In addition, the data management software and data files are often stored on only a single personal computer or computerized device. If the personal computer or other computerized device becomes lost or stolen, the user's data may no longer be accessible, and might end up in the possession of another person. If the personal computer or other computerized device crashes, the data can easily be lost.

[005] From the perspective of providers, such as vendors of on-line products or services, it can be valuable to have access to consumer information in order to, for example, facilitate e-commerce or m-commerce transactions, or else to better understand consumers or communicate with them about products or services in which they might be interested. However, consumers are often reluctant to provide their personal information, often in part due to concerns over security of the information. Also, consumers may not want to take the time to re-enter their personal information at different on-line provider sites. Providers of on-line products or services may therefore benefit from a mechanism which entices consumers to provide their personal information by minimizing the burden on consumers when conducting on-line transactions requiring personal information and by allowing consumers to retain control over the type and amount of information that is released to the provider.

[006] Accordingly, there remains a need for a more secure, flexible and convenient system for storing information and a method for allowing the user to manage and distribute that information using a personal computer or other network-connected device. There further remains a need for such a system and method that provides central information storage and does not require a permanent download of proprietary software to a client device for management and distribution of the

information. There is a need for a mechanism which encourages consumers to provide their personal information to providers of on-line products or services. Additionally, to facilitate the use of such a system, there is a need for a mechanism that provides consumers a method to conveniently and securely move to various web-sites without the need to repeatedly supply authentication information, such as username and password, etc.

Summary of the Invention

[007] According to certain aspects of the invention, a first request for access to the information account may be received by a host server from a network device. The host server may also receive authentication information from the network device in response to the consumer inputting the consumer authentication information while interacting with a first web-site. In response to receiving the consumer authentication information, the host server may authenticate the consumer, thereby allowing the consumer to access the information account. Then, the host server may communicate with the network device to automatically manage subsequent authentications of the consumer so that the consumer will not be required to again input the consumer authentication information upon initiating a second request for access to the information account while interacting with a subsequent web-site that is configured to provide access to the information account upon authentication of the consumer.

[008] The consumer may input a command for activating the single sign-on feature, i.e., the automatic management of subsequent authentications of the consumer. The single sign-on feature may involve determining that a previous authentication of the consumer for access to the information account remains valid and to instructing the subsequent web-site to by-pass a sign-on interface that would prompt the consumer to input the consumer authentication information when the consumer initiates the second request for access to the information account.

[009] In response to the first request for access to the information account, the network device may determine a first-determined equipment identifier that uniquely identifies the network device. The first-determined equipment identifier may be transmitted to the host server for storage in an authentication table in association with the consumer authentication information. The time of the sign-on may also be stored in the authentication table in association with the consumer authentication information. The host server may begin execution of the single sign-on

feature by recording in the authentication table in association with the consumer authentication information and the first-determined equipment identifier an indication that the single sign-on feature is activated.

[010] In response to the consumer initiating a second request for access to the information account, the network device may transmit to the host server a second-determined equipment identifier. In response to receiving the second-determined equipment identifier, the host server may consult the authentication table to determine whether the second-determined equipment identifier matches the first-determined equipment identifier. If so, the host server may determine from the authentication table whether the single sign-on feature is activated. If the single sign-on feature is activated, the host server may transmit to the network device a message that causes any sign-on interface associated with the information account to be by-passed. Prior to transmitting the message for by-passing the sign-on interface, the host server may determine whether a difference between a current time and the time at which the consumer was previously authenticated is not less than a time out interval or whether some other terminating event has occurred. If a terminating event has occurred, the message for by-passing the sign-on interface may not be sent.

[011] Additional embodiments, examples, variations and modifications are also disclosed herein.

Brief Description of the Drawings

[012] FIG. 1 is a high-level block diagram illustrating a system in accordance with one or more exemplary embodiments as disclosed herein.

[013] FIG. 2 is an abstract illustration of an information account in accordance with exemplary embodiments as may be used, for example, in the system illustrated in FIG. 1.

[014] FIG. 3 is an abstract illustration of another information account in accordance with other exemplary embodiments as may be used, for example, in the system illustrated in FIG. 1.

[015] FIG. 4 is an abstract illustration of an exemplary database schema in accordance with certain exemplary embodiments.

[016] FIG. 5 is a generalized interaction diagram illustrating the interaction between various system components of certain exemplary embodiments as disclosed herein.

[017] FIG. 6 is a generalized interaction diagram illustrating the interaction between various system components when a new information account is created by a consumer via a vendor's website, in accordance with one or more exemplary embodiments.

[018] FIG. 7 is a generalized interaction diagram illustrating the interaction between various system components in an exemplary wireless environment.

[019] FIG. 8 is a high-level block diagram illustrating logical grouping of vendor servers into exchanges in accordance with one or more exemplary embodiments as disclosed herein.

[020] FIG. 9 is an illustration of a web page displaying logos that identify a branded information account and exchange membership in accordance with one or more exemplary embodiments as disclosed herein.

[021] FIG. 10 is an abstract illustration of exemplary system components for implementing revenue sharing models in accordance with certain exemplary embodiments.

[022] FIG. 11 is a flow chart illustrating an exemplary single sign-on method in accordance with an exemplary embodiment of the present invention.

Detailed Description of Exemplary Embodiments

[023] In one or more embodiments, a system and method are provided for enabling consumers to store and maintain a comprehensive information profile (hereinafter "information account") in a centralized data repository that is accessible over a distributed electronic network, such as the Internet. The information account may be used to store any type of data desired by the consumer, including, for example, demographic information, financial information, medical information, family information, contact information, documents, image files, multimedia files, etc. The centralized data repository is preferably accessible via a network by any authorized network device. In various embodiments, no specialized application programs are required to be permanently downloaded to the consumer's network device in order to access the information account.

[024] According to certain embodiments, at the consumer's direction, selected information in the information account may be accessed and, if desired, shared with authorized vendors, business partners or any other entity that requires certain of the consumer's information. The terms "vendor" and "business partner" are

used herein in a general sense to refer to persons, businesses, enterprises or entities that make products or services available to consumers. As used herein, the terms "consumer," "buyer," and "user" are interchangeable.

[025] Server-side software or temporary client-side software may, in some embodiments, be used to manage communications with the information account and to automatically integrate that consumer information into a process executed by a network device. As an example, the network device may execute a business process relating to a consumer-initiated activity, such as a retail transaction. The server-side software or temporary client-side software may receive consumer information from the information account and use that information to automatically populate the input fields of a form or the input requirements of a process that is to be submitted to a vendor's server or other network device during an application, registration or transaction process.

[026] The data in the information account is preferably stored using a tagged data format. In one embodiment, the data in the information account may be stored using the eXtensible Markup Language ("XML") data format, which is an open standard for describing data from the World Wide Web Consortium ("W3C"). As is known in the art, XML tags are used to define the types of information that are represented by the data element. The XML standard provides a great deal of flexibility in that custom tags may be defined for any type of information that the consumer may desire to store in the information account. Using any well-known XML-related querying, parsing, transforming and/or filtering techniques, individual data elements in the information account may be accessed, updated, deleted, created, or otherwise manipulated.

[027] The information account may be structured as one or more data aggregates, e.g., XML data aggregates. An entire XML data aggregate is stored within a data field of a database table. This data field is a long text field containing all of the information associated with the given record. In one embodiment, all consumer information in the information account may be stored in a single XML data aggregate comprising consumer information elements and sub-elements. Attributes may also be associated with any element and sub-element in order to provide additional information. A transformation or filtering mechanism, such as "Style Sheets," may be applied to the single XML data stream in order to extract only selected data elements therefrom at the direction of the consumer.

[028] In an alternative embodiment, the information account may be normalized into a plurality of discrete data aggregates, each aggregate representing a predetermined “information product.” An information product refers to a package of consumer information relating to, for example, a specific product or service offered by a vendor or that is important to vendors with similar consumer information needs. For example, a mortgage information product might contain all consumer information that would be required to complete a lender’s mortgage application. Individual information products may be retrieved from the information account and transmitted to authorized vendors at the request of the consumer.

[029] Access constraints may be utilized in one or more embodiments as described herein to allow for the establishment of “exchanges.” An exchange generally refers to a group of entities that are authorized to accept consumer information from the information account at the request of the consumer. The information account may be accessed for retrieval of information to be used in commerce with any vendor or entity that is a member of the exchange. In much the same way that a consumer may have several different credit cards or debit cards that are each accepted only by certain merchants, the consumer may have several information accounts that are each valid only on specified exchanges.

[030] Exchanges may be implemented, for example, through “inflow” and/or “outflow” constraints imposed by the exchanges. An inflow constraint imposed by an exchange may, for example, dictate that only information accounts associated with specific other exchanges will be accepted or that no information accounts associated with other exchanges will be accepted. An outflow constraint may dictate that information accounts associated with an exchange may only be used within that exchange and within no other exchanges. Various business situations and partnerships may drive the implementation of inflow and outflow constraints. Revenue sharing models may be established in order to provide financial incentives to exchanges and/or individual vendors that facilitate the creation of an information account or the use of an information account to complete a transaction.

[031] Exemplary embodiments will now be described with reference to the drawings, in which like numerals represent like elements throughout the several figures. A high-level block diagram of a system in accordance with an exemplary embodiment is shown in and described with reference to FIG. 1. As shown, a central data repository 102 is provided for storing consumer information that may be easily

accessed from any network device attached to the network 106. The network 106 may comprise any telecommunication and/or data network, whether public or private, such as a local area network, a wide area network, an intranet, an internet and any combination thereof and may be wireline and/or wireless. Various methodologies as described herein may be practiced in the context of distributed computing environments. The network 106 thus provides for the open and seamless distribution of consumer information to and from the information account 110.

[032] In the system illustrated in FIG. 1, the exemplary operating environment encompasses various network devices for accessing and reading associated computer-readable media having stored thereon data and/or computer-executable instructions for implementing various methods of the present invention of data storage, management and distribution. Generally, a network device includes a communication device for transmitting and receiving data and/or computer-executable instructions over the network 106, and a memory for storing data and/or computer-executable instructions. A network device may also include a processor for processing data and executing computer-executable instructions, as well as other internal and peripheral components that are well known in the art (e.g., input and output devices.) As used herein, the term "computer-readable medium" describes any form of computer memory or a propagated signal transmission medium. Propagated signals representing data and computer-executable instructions are transferred between network devices.

[033] A network device may generally comprise any device that is capable of communicating with the resources of the network 106. A network device may comprise, for example, a network server 108 & 114, a client device 104, a wireless client device 104a or a dedicated storage device (e.g., the central data repository 102.) In the embodiment shown in FIG. 1, a host server 108 hosts the software for interacting with the central data repository 102 and for communicating with other network devices. The host server 108 may interact with the central data repository 102 via the network 106 or via a direct communication link 111. A vendor server 114 hosts vendor web page files 116 comprising a vendor website, through which products or services may be offered to consumers.

[034] A client device 104 may comprise a desktop computer, a laptop computer and the like. A wireless client device 104a may comprise a personal digital assistant (PDA), a digital and/or cellular telephone or pager, a handheld computer, or

any other mobile device. These and other types of client devices **104** & **104a** will be apparent to one of ordinary skill in the art. For convenience, the following explanation will be made with reference to a client device **104** generically, but, unless otherwise indicated, it will be understood that the principles and concepts described will also encompass wired or wireless devices, such as wireless client device **104a** illustrated in FIG. 1. Moreover, although exemplary embodiments will be described herein in the context of the Internet or a web-based environment, it will be appreciated that the various principles and methods of operation will be applicable or may be practiced in other environments as well.

[035] According to a preferred embodiment, a client device **104** may execute a browser **112** or another suitable application for interacting with web page files **116** hosted by a vendor server **114** and other network devices. Through the graphical user interface provided by a displayed web page file **116**, the vendor may require the consumer (i.e., the operator of the client device **104**) to input certain information pertaining to or associated with the consumer. According to certain embodiments, a consumer may be permitted to direct that the requested information be transmitted from the information account **110** to the client device **104** for processing. Although exemplary embodiments will be described herein in the context of a web-based environment, those skilled in the art will appreciate that other environments are suitable as well.

[036] The description of exemplary embodiments with reference to FIG. 1 assumes the existence of a previously created information account **110**. An example illustrating actual creation of an information account **110** will be described below with reference to FIG. 6. In general, the information account **110** may be any data structure for storing consumer information. Preferably, however, the information account **110** is stored as a tagged data structure, such as one or more XML data aggregates. The data in the information account **110** is preferably encrypted so that anyone gaining unauthorized access to the information account **110** will not be able to read the data. Also, in a preferred embodiment, each information account **110** in the central data repository **102** is encrypted separately, so that someone authorized to access the information account of one consumer may not also gain access to the information account of another consumer.

[037] In accordance with a preferred embodiment, the consumers may maintain sole responsibility for storing and updating the information in the

information account 110. Only the consumer, or those authorized by the consumer, may use the information account 110 to complete e-commerce or m-commerce activities. Consumers create an information account 110 either through a website hosted by the host server 108 or a website hosted by a vendor server 114. For example, after manually completing a form displayed by a vendor's website, the consumer can choose to create an information account 110 and have the consumer information stored therein.

[038] Upon creation of an information account 110, a consumer may be given an identification number, a username and/or a password. Other types of consumer authentication information are known in the art and may also be used in the context of the present invention. The system of FIG. 1 provides the consumer with a variety of methods of accessing the information account 110, transferring selected information to a vendor and/or allowing a vendor limited and constrained access to the information account 110, as described in further detail herein.

[039] A web page file 116 displayed by the browser 112 may include input fields for the input of consumer information. The web page file 116 may also include an instruction (e.g., a "call") that causes the browser 112 to download and execute a client-side application 105. JAVA applets are well known client-side applications and are particularly suited for use in various embodiments due to their platform-independent nature. However, any other type of client-side application may be used without departing from the spirit and scope of the present invention. The client-side application 105 resides in temporary memory storage of the client device 104, such as cache memory or the like, and may be removed from the client device 104 after its execution is complete. The client-side application 105 is specific to the browser session only and not to the client device 104. Multiple client-side applications 105 may be executed at the same time if multiple browser windows are executed by the client device 104. The client-side application 105 provides functionality for facilitating communications between the browser 112 executed by the client device 104 and the database management system ("DBMS") 109 of the host server 108.

[040] One responsibility of the client-side application 105 is to provide authentication information associated with the consumer and the vendor to the host server 108. Depending on the desired level of security within the system, authentication information may comprise a username, user ID, password, key, certificate and the like. Authentication information regarding the vendor may be

embedded within the web page file 116 for extraction by the client-side application 105. Alternatively, the client-side application 105 may communicate with the vendor server 114 to retrieve such vendor authentication information. Authentication information regarding the consumer may be supplied by the consumer via a user interface displayed by the client-side application 105 or by a displayed web-page file 116. Communications relating to authentication information may be accomplished using a secure transmission protocol or handshake, such as the secure shell BSD, Point to Point Tunneling Protocol (PPTP), also commonly known as Virtual Private Network, and/or secure socket layering (SSL) protocol. Other methods for achieving a secure connection over the network 106 will be apparent to those of ordinary skill in the art. Authentication information may also be encrypted and transmitted over an open network using any appropriate protocol.

[041] The client-side application 105 is also responsible for determining the type of consumer information that is required by the input fields of the displayed web page file 116. After determining the type of consumer information that is required, the client-side application 105 may formulate a database query in a language that is understood by the DBMS 109. At a minimum, client-side application 105 communicates enough information to the DBMS 109 regarding the required consumer information so that the DBMS can formulate a database query. In one embodiment, the DBMS 109 exposes an application program interface ("API") that can be utilized by the client-side application 105. An example of one such API is known as the Simple Object Access Protocol ("SOAP"). SOAP is a protocol that provides for interoperability between heterogeneous HTTP-based software and XML-based software. SOAP provides access to services, objects, and servers in a platform-independent manner. Since SOAP relies on HTTP as the transport mechanism, and most firewalls allow HTTP to pass through, SOAP endpoints may usually be invoked from either side of a firewall.

[042] The client-side application 105 may transmit the database query (or information to form the database query) to the host server 108 along with the above-mentioned authentication information over a secure connection. In such a scenario, the authentication information and the query information may be passed to the DBMS 109. The DBMS 109 attempts to authenticate the vendor and the consumer using the authentication information and corresponding information that was previously stored in the data repository 102. If authentication is successful, the DBMS 109 queries the

information account **110** using the appropriate database connectivity protocol, such as the Open Database Connectivity ("ODBC") protocol, the Java Database Connectivity Protocol ("JDBC"), or any other suitable protocol.

[043] As mentioned above, the data in the information account **110** may be encrypted. Thus, in response to the query, the DBMS **109** may receive an encrypted search result. The search result, for example, may be in the form of a stream of XML data that has been filtered from the information account. The DBMS **109** or other program module executed by the host server **108** may be responsible for decrypting the search result. The decrypted search results may then be transmitted to the client-side application **105** via the previously established or a new secure connection.

[044] In the alternative, the client-side application **105** may manage authentication and querying as separate processes. As an example, authentication may be handled using a secure connection as described above. Upon acknowledgment of authentication, the secure connection may be closed and the query process may be handled using open network communication protocols. In response to the query, the encrypted search result may be transmitted to the client-side application **105** over the open network and the client-side application **105** may be responsible for decryption.

[045] The client-side application **105** may also be responsible for parsing the data elements included in the search result and auto-populating the parsed data into the input fields of the displayed web page file **116**. Again, the client-side application **105** may translate the XML data into HTTP data using SOAP or another suitable protocol. Those skilled in the art will appreciate that in certain embodiments, especially where user verification of the consumer information is not required, the client-side application **105** may transmit the consumer information directly to the vendor server **114** without populating the consumer information into the displayed web page file **116**. If the input fields are auto-populated, the consumer has the opportunity to verify the information displayed in the input fields, make any necessary modifications, and then interact with the displayed web page file **116** to submit the information to the vendor server **114**. Any modifications to the consumer information that are made by the consumer may be detected by the client-side application **105**, which may then transmit the modified data back to the host server **108** for an appropriate update of the information account **110**. In addition, the client-side application **105** may determine whether the consumer inputs new data into the

input fields, and if so, transmit that new information to the host server **108** for storage in the data repository **102**. The consumer may interact with the displayed web page file **116** to submit the consumer information to the vendor server **114**. The vendor server **114** may then process the consumer information, as needed, by way of a processing module.

[046] In an alternative embodiment, a server-side application **107** may be employed instead of a client-side application **105** to manage communications with the host server **108**. An authorized server-side application **107** may receive consumer information directly from the host server **108** and present that consumer information to the client device **104** (e.g., via the browser **112**) for display to the consumer. A web page file **116** hosted by the vendor server **114** may be accessed and displayed by the browser **112** of the client device **104**. The displayed web page file **116** may present a user interface for input of consumer authentication information. In a preferred embodiment, the consumer authentication information is transmitted from the client device **104** to the host server **108** for authentication of the consumer. In addition, the client device **104** may also transmit a request that a “ticket” be provided to the vendor server **114**.

[047] As used herein, the term “ticket” refers to a temporary authorization for at least partial access to a consumer’s information account **110**. Although not shown in the figure, an information account **110** may be associated with a data table or other data structure that correlates one or more tickets with a set of consumer-defined attributes. The consumer-defined attributes may determine such things as the number of times that the password may be used to access the information account **110** (e.g., one-time use), any period of validity associated with the ticket (e.g., ticket expires one week from issuance), whether the ticket carries read, write and/or modify privileges, etc. The ticket attributes may also include any number of identifiers, such as a vendor identifier, a data identifier, and filter identifiers, which may be used to ensure that the party using the ticket is in fact authorized to do so, and to ensure that only authorized data is filtered for release to that party.

[048] Upon authenticating the consumer, for example by using standard browser authentication techniques, the host server **108** may redirect the browser **112** of the client device **104** to another web page data file **116** (e.g., another web page data file **116** hosted by of the vendor server **114**), including the ticket as a parameter in the URL. In response to detecting the ticket, the vendor server may extract the ticket and

pass it to the server-side application 107. The server-side application 107 may then use the ticket to authenticate itself to the host server 108, for example using SOAP or another suitable protocol.

[049] In accordance with one embodiment as described herein, a ticket generated by the host server 108 may be a "Globally Unique Identifier" ("GUID"). A GUID preferably comprises a unique number that is computed by adding the time and date to a network adapter's internal serial number, or by any other suitable technique. The ticket may be encrypted. For example, the ticket may be encrypted using the vendor's public key and the resulting binary encrypted blob may be base64 encoded so that it can be included as a parameter in a URL. At the vendor server 114, the parameter may be extracted from the URL, base64 decoded and then decrypted using the vendor's private key. Other encryption techniques may also be used.

[050] In an alternative embodiment, consumer authentication information may be submitted from the client device 104 to the server-side application 107 at the vendor server 114. The server-side application 107 may then transmit the consumer authentication information and vendor authentication information to the host server 108 for authentication of both the consumer and the vendor. The consumer authentication information may be encrypted at the client device 104 and decrypted only at the host server 108. Such an embodiment, however, places a significant amount of control over the consumer's data in the hands of the vendor, and thus may not be preferable.

[051] The server-side application may be identified by an application identifier ("APPID"). The APPID may be associated at the host server 108 (e.g., by the DBMS 109) with a particular filtering mechanism. As mentioned, style sheets are well-known and highly suitable filtering tools for use in conjunction with XML data. In response to authenticating the server-side application 107 and identifying the appropriate filter, consumer information may be filtered from the information account 110 and transmitted back to the server-side application 107. The server-side application 107 may then parse the consumer information, for example, in order to auto-populate a form, which may or may not have been previously displayed to the consumer.

[052] As in the case of the client-side application 105, the server-side application 107 may receive decrypted consumer information from the host server 108 via a secure connection, or may receive encrypted consumer information via the

open network. Thus, the server-side application **107** may be configured to perform decryption as necessary. The consumer information thus received from the host server **108** may be presented to the consumer for verification. Any modifications or additions made to the consumer information may be submitted back to the server-side application **107** for communication to the host server **108**. The DBMS **109** may then update and/or create the information account **110** in the appropriate manner. The consumer may interact with the displayed web page file **116** to submit the consumer information to the vendor server **114**. The vendor server **114** may then process the consumer information, as needed, by way of a processing module.

[053] Those skilled in the art will appreciate that the illustration and discussion of exemplary embodiments with reference to FIG. 1 is provided as a generalized example only. Specific details regarding data formats and network communication protocols have been omitted, as such details are well known in the art. Furthermore, the present invention is not intended to be limited to the use of any particular data formats or protocols. Any existing or future formats or protocols may be used without departing from the spirit and scope of the invention. Furthermore, many network components were not shown or discussed with reference to FIG. 1, such as gateways, routers, hubs, switches, firewalls, DNS servers, authentication servers, certificate authorities, and the like. The functions and roles of such network components are also well known in the art and need not be described in detail herein.

[054] FIG. 2 provides an abstract illustration of an information account **110** in accordance with an exemplary embodiment as described herein. In the illustrated embodiment, the consumer information is stored in the information account **110** as a single tagged (delimited) data stream. XML generally provides a suitable tagged data format; however, other tagged data formats can be employed as well. Thus, references to the XML standard in connection with exemplary embodiments are not intended to limit the scope of the present invention. The single XML data stream comprises a plurality of consumer information elements **202**, each having a unique tag **204** or identifier. A consumer information element **202** may be divided into any number and/or level of sub-elements **206**. As is well known in the art, an XML consumer information element **202** may also be associated with one or more attributes **208**. An attribute **208** may provide additional information about the content, structure or formatting of a consumer information element **202**.

[055] A consumer information element **202** may comprise any type of data or information, including text strings, objects, files, applications, etc. Obviously, the more consumer information that is stored in the information account **110**, the larger the XML data stream will be. The size of the XML data stream is limited only by the hardware and software limitations of the system (e.g., memory size, processor speed, bandwidth, etc).

[056] An information account **110** is preferably unique to a single customer. Each information account **110** stored in the data repository **102** may thus comprise a discrete XML data stream. Each information account **110** stored in the data repository **102** may be individually encrypted. For example, one method for encrypting an information account **110** may involve use of the consumer's public key. Accordingly, only someone having access to the consumer's private key will be able to decrypt the consumer's information. Many other and/or additional methods for encrypting information accounts **110** and/or the entire data repository **102** will occur to those skilled in the art.

[057] Although not shown in FIG. 2, those skilled in the art will appreciate that a consumer information element **202** in one information account **110** may comprise a pointer or a reference to another data element or to another information account **110**. In one embodiment, a consumer may create, for example, a list of business contacts. A new information account may be created for each individual specified as a business contact by the consumer. Authentication data within the new information account may be set as "anonymous" so that the first consumer may retain access privileges. At some point later, however, the individual named as the business contact may be given control of the new information account by changing the associated authentication information to be unique to that individual. The first consumer may then be granted limited access privileges to continue to access the new information account of the business contact (e.g., by way of a ticket). Alternatively, the first consumer may retain a copy of the business contact information in his own information account.

[058] FIG. 3 provides an abstract illustration of an information account **110** in accordance with other exemplary embodiments. In the embodiment shown, an information account **110** is structured as multiple discrete XML aggregates **302a-c**. The discrete XML aggregates **302a-c** may comprise one primary "profile" record **302a** and one or more information product records **302b-c**. The profile record **302a**

may include a general profile of information elements **304** associated with the consumer. Information product records **302b-c** contain consumer information elements that, for example, are specific to a particular product or service offered by a vendor or that are important to vendors with similar consumer information needs. Aggregation of data elements according to information products allows quick and efficient retrieval of specific consumer information from the information account **110** through a request-response system.

[059] The number of aggregates or records included within the information account **110** of a given consumer depends upon the number of information products for which the consumer has elected to store information. For example, a consumer who has elected to store information about two separate products, such as a car loan and a mortgage loan, would have at least three data aggregates in his information account **110**. One such data aggregate would represent the primary profile record and each of the two other data aggregates would include information about one of the information products. Data aggregates may include but are not limited to the following information products: Home Loan, Auto Loan, Student Loan, Home Insurance, Auto Insurance, Life Insurance, Online Banking, Credit Card, Government Services, Education, Career, Travel, Retail, and Relocation. If a consumer creates or updates an information account via a vendor's web site and thereby inputs information regarding a new product, a new product record **302b-c** will be created in the information account. Each product record **302b-c** created for the consumer is of course associated with the primary profile record **302a**.

[060] If an information account **110** is segmented into multiple discrete data aggregates, there may be a need for maintaining consistency among redundant data elements stored in multiple information products. "Latent referential processing" is one method for maintaining data consistency, and in this context refers to the use of a series of pointers or references to flag data that is redundant across multiple products. According to latent referential processing, when a record **302a-c** is created or updated, redundant information elements that are stored in other data aggregates typically are not also updated until the next time the information account is accessed. For example, if salary information is updated in a home loan information product record, redundant salary information in the consumer's auto loan information product record will generally not be immediately updated. Thus, latent referential processing allows data inconsistencies to exist within the information account after an update.

[061] As is shown and described with reference to FIG. 4, a transaction log (e.g., a time stamp log) may be maintained for each redundantly stored aggregate in the information account to record the date and time of the most recent update for each data record **302a-c**. Each time a request is made to access the information account, the DBMS **109** may first examine the time stamp log to determine which data element in a set of redundant data elements has most recently been updated. After determining the most recently updated data element, all other redundant data elements are updated to be consistent with the most recently updated data element. Upon completion of the latent referential processing, the request to access the information account may be granted. Accordingly, latent referential processing is a new way of storing and tracking information that addresses the need of providing quick access to information that will be accessed more frequently than it will be updated.

[062] In another embodiment, redundancy and consistency concerns are addressed by normalizing the data aggregates of the information account **110** to the extent possible. For example, an information account **110** may be configured such that the consumer's profile record **302a** stores the majority of the consumer's personal information. The profile record **302a** may comprise predefined data elements, such as "first name," "middle name," "last name," date of birth," etc. The profile aggregate **302a** may also be expanded to include any additional and/or custom fields. Additional aggregates corresponding to information products **302c** may contain pointers **306** to the data fields within the profile aggregate **302a**. Thus, the information account **110** may be configured to store within one aggregate a single instance of an information element that is referenced by other aggregates. As information product aggregates **302c** are formed independently of the profile aggregate **302a**, data elements that are not unique to those information product aggregates **302c** may be ported into the profile aggregate **302a** if desired.

[063] FIG. 4 illustrates an exemplary database schema **400** in accordance with one or more exemplary embodiments as disclosed herein. In particular, the database schema **400** represents the situation where the information account **110** is segmented into multiple discrete data aggregates, as shown in FIG. 3. The database schema **400** may include a consumer authentication record **402** that stores consumer authentication information **404** such as, for example, a user ID, username, password, email address, access attempts, last attempt date/time, challenge word or phrase, challenge response, ticket parameters, and vendor credited with origination of the

information account. These and other types of authentication information may be used to authenticate a consumer. The database schema **400** may also include a profile record **302a** that stores a primary information profile **304** of the consumer. There will typically be a one to one relationship between the consumer authentication table **402** and the profile record **302a**. The exemplary database schema **400** also includes one or more information product records **302b-c** that store product-specific information. Each profile record **302a** may be associated with one or many information product records **302b-c**.

[064] The profile record **302a** and each information product record **302b-c** may further be associated with a transaction log record **406**. Each time the profile record **302a** or an information product record **302b-c** is acted upon, detailed transaction information **408** may be recorded in a new transaction log record **406** (not to be confused with the above-mentioned time stamp log.) Transaction information **408** may provide the basis for all transaction billing and revenue sharing events. By way of example only, the transaction record **406** may identify the vendor server through which the information account **110** was created. The transaction record **406** may also identify the vendor server through which a transaction was completed using the information account **110**.

[065] As used herein, the term "transaction" refers broadly to any activity related to an information account, including, but not limited to a *create* transaction, *delete* transaction, *update* transaction, *authentication* transaction, a request for information from authorized vendors, a client device and/or vendor server **114** request, a publishing and form filling transaction, and a *submit* transaction where the information account **110** is processed into the requesting vendors systems. A portion of any monies billed upon completion of a transaction may be shared with each of the vendor servers identified in the transaction record **406**.

[066] FIG. 5. is a generalized interaction diagram illustrating the interaction between various system components of certain exemplary embodiments in connection with consumer-controlled storing, managing and/or distributing information. The exemplary embodiments discussed with reference to FIG. 5 employ a client-side application **105**, such as an applet, to manage communication between the client device **104** and the host server **108**. Alternative embodiments employing a server-side application **107** instead of the client-side application **105** have been

discussed above. Those skilled in the art will appreciate the differences between the interactions involving a client-side application 105 and a server-side application 107.

[067] The generalized interaction diagram begins at step 501, where the consumer operates a browser 112 to retrieve a web page file 116 from the vendor server 114 via the network 106, using a consumer browser. The web page file 116 retrieved from the vendor server 114 may be enabled for interaction with the consumer's information account 110 and may thus include an instruction that causes the browser 112 to download a client-side application from the host server 108. At step 502, the client-side application is downloaded from the host server 108 to the browser 112. At step 504, the consumer interacts with the browser 112 to request use of the information account 110, which in this example has already been created. The web page file 116 may display a selectable icon or other indicia that allows the consumer to request use of the information account 110. Alternatively, the client-side application 105 may provide the interface for requesting use of the information account 110.

[068] Next at step 506, the client-side application 105 displays a login interface to the consumer. The login interface may be displayed, for example, in the open display window of the browser 112, in a pop-up window, or in any other suitable manner. At step 508 the consumer inputs consumer authentication information, which is transferred from the browser to the client-side application 105. Consumer authentication information may comprise, for example, a username, user ID, password, challenge phrase, email address, etc. At step 510, the user authentication information is combined with vendor authentication information and is sent to the DBMS 109. Vendor authentication information may comprise a vendor ID, password, product IP, application ID, and the like. Vendor authentication information may be used to authenticate the vendor and to determine the manner in which consumer information is to be filtered from the information account 110.

[069] After the DBMS 109 receives the authentication information, it submits an authentication request to the data repository 102 at step 512. The authentication request may be a database query to determine if the supplied consumer authentication information and vendor authentication information are consistent with previously stored authentication information. In response to authenticating the consumer and the vendor, the DBMS 109 performs one or more database queries at step 514 to retrieve consumer information elements from the information account

110. Depending on the structure of the information account, the DBMS **109** may retrieve certain products (identified by product ID) from the information account **110**, or may retrieve a set of data elements filtered according to a vendor ID or an application ID. If consumer information is retrieved according to products, an iterative lightweight transfer (“LWT”) process may be performed in order to get the best set of data elements for each new product ID. Lightweight transfer techniques are well-known in the art and generally involve the use of thin protocols and/or smart proxies that can cache results and perform buffered reads and writes, minimizing the number of network calls.

[070] Once the DBMS **109** has retrieved the relevant consumer information, the consumer information elements may be merged (if appropriate) decrypted (if appropriate) and/or further filtered (if appropriate) at step **518**. Then, at step **520**, the resulting information elements are transmitted to the client-side application **105**, for example in the form of an XML data stream. At step **522**, the client-side application **105** parses the received XML data and transforms it into the required format for populating the input fields of the displayed web page file **116**. The client-side application **105** then auto-populates the input fields of the displayed web-page file **116** at step **524**. The consumer may interact with the browser **112** to edit or modify the auto-populated information at step **526**. Because there may be multiple web page files **116** associated with the vendor website, steps **524** and **526** are repeated until all data has been auto-populated and/or edited on every included web page. The client-side application **105** monitors the edit process to determine if the consumer desires to modify and/or supplement any of the consumer information elements.

[071] The consumer may then interact with the browser **112** at step **528** in order to submit the consumer information that has been entered into the displayed web page file(s) **116** to the vendor server **114**. The vendor server **114** receives and processes the consumer information elements at step **530**. After processing the consumer information, the vendor server **114** preferably transmits a “success page” or other acknowledgement to the consumer’s browser **112** at step **532**.

[072] Either through a selectable icon or other indicia displayed on the success page or displayed by the client-side application **105**, or any other interactive means, the consumer may interact with the browser **112** at step **534** to submit an update request to the DBMS **109**. Update is an event whereby the information account **110** is updated to reflect any edits that the consumer may have made to the

consumer information at step 526. Thus, a consumer is permitted to update the information account 110 via a vendor's website. As another option, the consumer may elect to update the information account 110 at a later time directly via the host server 108.

[073] At step 536 the client-side application submits the consumer's XML data (possibly only the edited data) and the update request to the DBMS 109. Then at step 538 the update request is submitted to the data repository for authentication. In the authentication process, consumer authentication information, vendor authentication information and, if appropriate, product identification information (which are all included in the update request) are verified. Upon authentication of the update request, the XML data is validated at step 540 and the update is performed at step 542. The DBMS then sends the update result (success or failure) to the client-side application 105 at step 544, which in turn displays the update result to the browser 112 at step 546. The exemplary generalized interaction diagram then ends at step 548.

[074] FIG. 6 is a generalized interaction diagram illustrating the interaction between main system components when a new information account is created by a consumer via a vendor's website. As mentioned, the consumer may create an information account by visiting a vendor's website that has been configured to allow creation of an information account. The vendor's website may, for example, require the user to manually input consumer information into the input fields of a form. The user may then direct that an information account be created to store the consumer information, so that the consumer will not be required to manually enter the consumer information again on any participating website.

[075] The exemplary embodiments discussed with reference to FIG. 6 employ a client-side application 105, such as an applet, to manage communication between the client device 104 and the host server 108. Alternative embodiments employing a server-side application 107 instead of the client-side application 105 have been discussed above. Those skilled in the art will appreciate the differences between the interactions involving a client-side application 105 and a server-side application 107.

[076] The exemplary interaction diagram of FIG. 6 begins at step 601, where the consumer operates a browser 112 to retrieve a web page file 116 from the vendor server 114 via the network 106, using a consumer browser. The web page file 116

retrieved from the vendor server 114 may be enabled for interaction with the consumer's information account 110 and may thus include an instruction that causes the browser 112 to download a client-side application from the host server 108. At step 602, the client-side application is downloaded from the host server 108 to the browser 112. At step 604, the consumer interacts with the browser 112 to input consumer information into the input fields of the vendor's website. The client-side application 105 monitors the input of consumer information at step 606.

[077] Next at step 608 the consumer interacts with the browser 112 in order to submit the consumer information to the vendor server 114. The vendor server 114 receives and processes the consumer information elements at step 610. After processing the consumer information, the vendor server 114 transmits a "success page" or other acknowledgement to the consumer's browser 112 at step 612. Either through a selectable icon or other indicia displayed on the success page or displayed by the client-side application 105, the consumer may interact with the browser 112 at step 614 to submit a request for creation of an information account 110 to the DBMS 109. Thus, the consumer may be permitted to create an information account 110 via a vendor's website. As another option, the consumer may elect to create an information account 110 at a later time directly via the host server 108.

[078] At step 616 the client-side application submits the consumer's XML data and the create request to the host server 108. Then at step 618 the host server 108 transmits an information account creation interface to the browser 112. The consumer inputs consumer authentication information via the information account creation interface at step 622 and the browser 112 passes the create request (which may include the consumer authentication information, the vendor authentication information, etc.) to the client-side application 105 at step 624.

[079] At step 626, the create request is combined with the consumer's XML data and is sent to the DBMS 109. In response to receiving the authentication information, the DBMS 109 submits an authentication request to the data repository 102 at step 628. The authentication request may be a database query to determine if the supplied consumer authentication information and vendor authentication information are consistent with previously stored authentication information. In response to authenticating the consumer and the vendor, the DBMS 109 validates the consumer's XML data at step 630 and creates a new information account 110 at step 632.

[080] Once the information account has been created, the DBMS 109 sends the create result (success or failure) to the client-side application 105 at step 634, which in turn displays the create result to the browser 112 at step 636. At step 638, the host server 108 creates an acknowledgment email to be sent to the consumer's email account. At step 640, the host server requests and receives the consumer's email address from the DBMS 109. At step 642 the consumer's acknowledgment email is delivered to the consumer. The exemplary generalized interaction diagram then ends at step 644.

[081] FIG. 7 is a generalized interaction diagram illustrating the interaction between various system components in an exemplary wireless environment suitable for implementation of systems or methods for consumer-controlled storage, management and/or distribution of information. An exemplary wireless environment is suited for wireless devices such as digital or cellular telephones, personal digital assistants ("PDAs"), portable computers, and the like. Such wireless devices generally include a display device and an input device (keypad, touch screen, microphone, etc.), each of limited size and utility. The difficulty of inputting detailed information and commands into a wireless device makes it desirable to provide a system whereby the backend DBMS 109 is able to communicate directly with various remote web servers, thus eliminating a significant amount of user-interaction with the wireless device.

[082] The generalized interaction diagram of FIG. 7 begins at step 701, where the consumer operates a wireless client device 104a to access the host server 108. Accessing the host server 108 may involve, for example, calling a dedicated access number using a mobile telephone device or two-way pager. At step 702, the wireless client device 104a accesses the host server 108 via a wireless application ("WAP") gateway. At step 704, the host server 108 returns a login interface to the wireless client device 104a. At step 706 the consumer inputs consumer authentication information using an input device of the wireless client device 104a. Consumer authentication information may comprise, for example, a username, user ID, password, challenge phrase, email address, etc.

[083] At step 708, the user authentication information is combined with vendor authentication and is sent to the DBMS 109. Vendor authentication information may comprise a vendor ID, password, product IP, application ID, and the like. Vendor authentication information may be used to authenticate the vendor and

to determine the manner in which consumer information is to be filtered from the information account **110**. After the DBMS **109** receives the authentication information, it submits an authentication request to the data repository **102** at step **710**. In response to authenticating the consumer and the vendor, the DBMS **109** performs one or more database queries to retrieve consumer information elements from the information account **110**. Depending on the structure of the information account, the DBMS **109** may retrieve certain products (identified by product ID) from the information account **110**, or may retrieve a set of data elements filtered according to a vendor ID or an application ID. If consumer information is retrieved according to products, an iterative lightweight transfer ("LWT") process may be performed at step **712** in order to get the best set of data elements for each new product ID. Otherwise, the consumer information elements are retrieved from the data repository **102** using appropriate filters at step **714**.

[084] Once the DBMS **109** has retrieved the relevant consumer information, the consumer information elements may be merged (if appropriate), decrypted (if appropriate) and/or further filtered (if appropriate) at step **716**. Then, at step **718**, the resulting information elements are transmitted to the vendor server **114**, for example, in the form of an XML data stream. The vendor server **114** receives and processes the consumer information elements at step **720**. After processing the consumer information, the vendor server **114** transmits a delivery receipt acknowledgment to the host server **108** at step **722**. The host server **108** may then pass an acknowledgment (success or failure) to the consumer (e.g., to the wireless client device **104a** or to another client device **104**) at step **724**. The exemplary generalized interaction diagram then ends at step **726**.

[085] As shown in FIG. 8, information accounts **110** may be used in the context of one or more exchanges **802A&B**. In this context, an exchange **802A&B** may comprises a group of entities (e.g., vendor servers **114**) that are authorized and configured to accept consumer information from a particular information account **110** at the request of the consumer. An information account **110** may, in some embodiments, be used to retrieve information for use in commerce with any vendor that is a member of the exchange **802A&B**. An information account **110** may be accepted in one or more exchanges **802A&B** according to various rules and relationships, as illustrated by the examples set forth herein. A consumer may also

have several different information accounts **110**, each valid for use in one or more exchanges.

[086] An exchange may comprise a logical grouping of servers or other network devices, and those skilled in the art will appreciate that there are a variety of suitable methods for implementing logical groupings of network devices on a distributed network. For example, an exchange identifier may be used to identify an exchange and may be associated with each network device that is a member of that exchange. In such an embodiment, look-up table of exchange identifiers may be maintained at the host server **108**, within the central data repository **102** or at another suitable location and may be used to authenticate an exchange identifier used in connection with a request for access to an information account **110**.

[087] Exchanges **802A&B** may be implemented, for example, through inflow and/or outflow constraints. An inflow constraint may, for example, dictate that only information accounts **110** associated with specific other exchanges will be accepted within an exchange or that no information accounts **110** associated with other exchanges will be accepted. An outflow constraint may dictate that information accounts **110** associated with an exchange may be used within that exchange and within no other exchanges (i.e., a private exchange), or within only selected other exchanges. Various business situations and partnerships may drive the implementation of inflow and outflow constraints.

[088] In various embodiments, an information account **110** may be branded so as to be associated with a particular vendor or other entity, product or service. By way of example only, if a consumer creates an information account **110** via a website maintained on behalf of a particular vendor, e.g., "Vendor X," the information account **110** may be branded as a "BrandX" information account **110X**. A BrandX information account **110X** may be stored in the central data repository in association with a BrandX identifier. BrandX logos or indicia may be displayed to the consumer when the consumer accesses the BrandX information account **110X**. Thus, although Vendor X "sponsors" the BrandX information account **110X**, the central data repository **102** that stores the BrandX information account **110X** may be maintained by another entity.

[089] An exchange **802A&B** may be configured to accept one or more differently branded information accounts **110**. This concept is similar to automated teller machine (ATM) networks, in which a customer of one bank may use his ATM

card (e.g., debit or credit card) to conduct transactions at the ATM of another bank. Typically, an ATM card includes a number of logos (also referred to as “bugs”) that indicate the financial networks that will accept the ATM card. ATMs also display logos identifying the financial networks to which they are connected. Thus, a bank customer may have a Wachovia® ATM card that is accepted in all Honor and PLUS network ATMs. Similarly, the various vendor servers **114** that make up a particular exchange may include logos or other indicia indicating the brands of information accounts **110** that will be accepted.

[090] With reference to FIG. 8 and FIG. 9, a consumer interacting with a browser **112** of a client device **104** may be presented with a web page file **116Y** by a vendor server **114Y** maintained by Vendor Y. The displayed web page file **116Y** may display an enrollment application link **902** that, when selected, will cause an enrollment application to be presented to the consumer. An enrollment application may be a form or other interface that prompts the consumer to input selected information. The website of Vendor Y may be configured, as described above, for interaction with the central data repository **102** via the host server **108**. Furthermore, the vendor server **114Y** may be a member of “Exchange B” **802B** that also includes vendor server Z **114Z**. For the sake of example only, it may be assumed that the inflow constraints of Exchange B **802B** allow any member vendor server (**114Y&Z**) to accept BrandY information accounts **110Y**, BrandZ information accounts **110Z** and BrandX information accounts **110X**.

[091] The displayed web page file **116Y** may thus display one or more brand logos **904** indicating the accepted brands of information accounts. The displayed web page file **116Y** may also display one or more exchange logos **906** indicating the exchanges of which the vendor server **114Y** is a member. In addition, the displayed web page file **116Y** may display an access/create link **908** for allowing a consumer to access or create a BrandY information account **110Y**. The displayed web page file **116Y** of FIG. 9 is shown by way of example only and that may other arrangements are possible. In perhaps a more practical example, the brand logos **904**, the exchange logos **906** and the access/create link **908** might be presented to the consumer only if the consumer selects the enrollment application link **902**. Other types of user interfaces may also be used.

[092] When used in the context of a private exchange (e.g., an exchange that does not accept foreign information accounts **110**) an information account may take

the form of a "private" branded information account **110**. As an example, if Vendor X establishes a private Exchange A **802A** that offers a variety of financial services, a BrandX information account **110X** may be established for consumers who participate in the private exchange. The BrandX information account **110X** may be configured to store information that is relevant to the financial services offered by Vendor X. If appropriate outflow constraints are established, the BrandX information account **110X** may be accepted only within private Exchange A **802A**. Again, Vendor X may facilitate or otherwise sponsor the creation of the BrandX information account **110X**, while another entity may server as the custodian of the data repository **102** for storing the BrandX information account **110X** and provide the underlying information technology.

[093] If private Exchange A **802A** is not subject to outflow constraints, a BrandX information account **110X** may also be accessed at websites hosted by or on behalf of other vendors, such as Vendor Y and/or Vendor Z. Consequently, an on-line form associated with Vendor Y web page files **116Y** or Vendor Z web page files **116Z** may automatically be populated based on information elements originating from the BrandX information account **110X**. Similarly, if Exchange A **802A** is subject to appropriate inflow constraints, a BrandY information account **110Y** and a BrandZ information account **110Z** may also be used at any website hosted by a vendor server **114X** that is a member of the Exchange A **802A**. In general, any number of vendors or other entities may participate in an exchange.

[094] Various licensing arrangements and revenue sharing agreements may be established between the custodian of the data repository **102** and the vendors that configure their vendor servers **114** for interaction with information accounts **110**. In particular, the custodian may choose to implement revenue sharing models in order to provide vendors with an incentive to promote and facilitate the creation and use of information accounts **110**. The custodian may earn revenues in exchange for the service of providing access to information accounts **110** for completion of transactions. For example, the custodian may be paid a per transaction commission by the requesting exchange or vendor each time an information account **110** is used by a consumer to quickly fill out a form or other document for completing a transaction with a vendor. As another example, the custodian of the data repository **102** may receive revenue from the requesting exchange or vendor based on milestone transaction numbers. For example, the custodian may be paid a negotiated dollar

amount for a negotiated number of transactions (e.g., \$100 for every 500 transactions completed using an information account).

[095] The more information accounts **110** that are in existence, the more transactions that are likely to occur in commerce. Accordingly, the custodian of the data repository **102** may choose to implement various revenue sharing models in order to financially encourage vendors and other entities to promote and/or sponsor information accounts **110**. As an example, a revenue sharing model may specify that a lifetime revenue stream be paid to the originating vendor or entity that is credited with facilitating the creation of an information account **110**. A lifetime revenue stream may be effective for the life of the information account **110** and may take the form of a credit issued to the originating vendor or entity each time that information account **110** is used to complete a transaction. A credit may amount to a percentage (anywhere from 0% to 100%) of the revenue earned by the custodian of the data repository **102** in connection with the transaction, or an otherwise arranged fee. Revenue sharing models may also specify that credits be paid by the custodian of the data repository **102** to a transacting vendor or entity that accepts consumer information elements from an information account **110** in order to complete a transaction.

[096] In the context of exchanges and branded information accounts, the amounts credited to originating entities and transacting entities may vary depending on the particular exchange and/or which brand of branded information account was used in order to complete a transaction. For example, referring back to FIG. 9, the custodian of the central data repository **102** may grant larger credits to a transacting vendor (Vendor X) when a BrandY information account **110Y** (that is, an information account from another exchange) is used to complete a transaction through the vendor server **114X**, as opposed to when a BrandX information account **110X** (that is, an information account from the same exchange) is used to complete a transaction through the vendor server. As mentioned, any number of factors or business relationships may affect the revenue sharing models adopted by the custodian of the central data repository **102**. As will be appreciated by those of skill in the art, different and/or multiple revenue sharing models may be applied to different exchanges or associated with differently branded information accounts. Members of an exchange may also choose to establish their own additional revenue sharing

models, for example, in an attempt to maximize the acceptance of a branded information account.

[097] Revenue sharing models may further include credits paid to OEMs, consultants, software providers and/or any other party who facilitates the creation and/or construction of an exchange, introduces information accounts **110** to an exchange, or otherwise assists the custodian of the central data repository **102** in increasing its revenue base.

[098] FIG. 10 is an abstract illustration of system components for implementing revenue sharing models in accordance with certain exemplary embodiments as disclosed herein. As shown, the central data repository **102** may store one or more transaction logs **1002** containing information relevant to any transaction that involved an information account **110**. The transaction log **1002** may identify, for example, the date, time and nature of the transaction, the originating entity, the transacting entity, whether the information account **110** was branded, etc. Many alternatives for storing and identifying transaction information are possible in the context of the illustrated embodiment. For example, each information account **110** may include or have associated therewith a unique transaction log **1002**. Alternatively, a transaction log **1002** may be used to store transaction information associated with multiple information accounts **110**.

[099] An extraction module **1004** may be used to facilitate the extraction of transaction information from a transaction log **1002**. The extraction module **1004** may be executed by the host server **108** or by another network device that is in communication with the host server **108** or the central data repository **102**. The extraction module **1004** may be employed to extract selected transaction information from the transaction log **1002** and to translate or transform the extracted transaction information into a format that can be interpreted by a financial processing system **1006**. Thus, in certain embodiments, the extraction module **1004** may be configured to extract transaction data elements from a tagged data stream representing or associated with an information account **110**. SOAP and/or other well-known protocols may be used by the extraction module **1004** to interface between the transaction log **1002** and the financial processing system **1006**. The financial processing system **1006** may comprise any system for processing transaction information and revenue sharing models in order to ensure that the appropriate party is billed in connection with a transaction involving an information account and that

revenues are shared with the appropriate parties. By way of example only, the financial processing system may be a custom software module or an off-the-shelf software package, such as the well-known "Oracle Financials" package.

[0100] Those skilled in the art will appreciate that the system components and arrangement thereof shown in FIG. 10 are by way of example only. Various other methods for recording and processing transaction information may be used in accordance with the concepts and principals discussed or suggested herein.

[0101] In connection with the creation of an information account 110, a consumer may be provided with consumer authentication information, which may include, for example, a username, password, user ID, biometric, challenge word, phrase or response, etc. This consumer authentication information may be stored in the consumer's information account 110, along with other authentication-related information such as, for example, email address, access attempts, last attempt date/time, challenge query, ticket parameters, vendor credited with origination of the information account, etc. In certain embodiments as disclosed herein, a single sign-on mechanism (also referred to herein as a single sign-on feature) may be provided to allow a consumer to "sign-on" (i.e., to provide consumer authentication information as may be required) for authentication to securely access an information account 110 at a first website. Since a consumer's information account 110 may be accessible from more than one website, the authentication status may be handled in such a way so as to "follow" the consumer as the consumer accesses subsequent websites. At such subsequent websites, a consumer who has activated the single sign-on mechanism need not re-enter authentication information, assuming certain conditions are present.

[0102] A preferred single sign-on mechanism can be implemented, in certain embodiments, without requiring a manual download or installation of any program modules on the consumer's client device 104. Nor does the single sign-on mechanism, at least in a preferred embodiment, require "add-ons", "cookies" or other special configurations for a web browser, although such features may optionally be utilized in connection with or in addition to a single sign-on mechanism as disclosed herein. A preferred single sign-on mechanism is managed at the client device 104 via one or more client-side applications 105 that are loaded into the browser 112 along with web page files 116 that comprise a consumer information account-enabled website. Applets (e.g., JAVA applets) are particularly well-suited for use as client-

side applications **105** in this context, due to their platform independent nature. In an exemplary embodiment of the single sign-on mechanism, a client-side application **105** (e.g., applet) may communicate with the host server **108** to determine whether the user has already been authenticated, and if so, to cause the log-in interface to be bypassed. Re-authentication may thereby be performed automatically by way of the client-side application **105**.

[0103] FIG. 11 is a flow chart illustrating an exemplary single sign-on method. The method begins at step **1101**, whereupon a consumer using a client device **104** downloads an information account-enabled web page file **116** which is displayed by browser **112**. The web page file **116** may include an instruction (e.g., a "call") that causes the browser **112** to download and execute one or more client-side applications **105**, which may be used to manage, among other things, the general request/response process involved in accessing and retrieving information from an information account **110** for the client device **104**. Client-side application(s) **105** may further be used to implement and manage functions of the single sign-on feature at the client device **104**. Those skilled in the art will appreciate that management of single sign-on functions may be performed by the same or different client-side application(s) **105** that manage the general request/response process.

[0104] After the client device **104** has downloaded the information account-enabled web page file **116**, the user may activate an access/create link **908** of the consumer information account-enabled displayed web page file **116** and, assuming that the single sign-on feature is not activated, may subsequently supply consumer authentication information (e.g., username/password, etc.) via a sign-on interface in order to request access to an information account **110**. At step **1102**, the client-side application **105** responsible for authentication receives the consumer authentication information supplied by the user. Then, at step **1104** the client-side application **105** determines a browser identifier that uniquely identifies the browser from which the sign-on request was initiated and the sign-on time (i.e., the time the sign-on request was initiated.) The browser identifier may comprise any unique identification code, such as a product serial number (relating to hardware or software), a dynamically generated alphanumeric string, etc. The sign-on time may be determined, for example, by interacting with a clock function executed by the client device **104**. It is expected that the client device **104** (a personal computer, for example, has a system clock from which the current time may be read. However, it is also possible that to

obtain the current time from a remote site across the network **106**. The sign-on time may be stored as wither an absolute time value, or else as a relative time value with respect to a known reference time.

[0105] Those skilled in the art will appreciate that any equipment identifier that uniquely identifies the client device **104** may be substituted for the browser identifier. For example, mobile client devices **104a**, such as network-enabled telephones, PDA, portable computers and the like may be assigned unique equipment identifiers, which may be static or dynamic. A client-side application **105** may thus be configured to determine any unique equipment identifier and to transmit that unique equipment identifier to the host server **108**. Furthermore, an equipment identifier may be generated or determined at the network device **104** or may be received from another source, such as the host server **108**, a certificate authority or some other authentication entity. Accordingly, any reference herein to a browser identifier is intended merely to provide an example of certain embodiments of the present invention and is not intended to limit the scope thereof.

[0106] The order in which the client-side application **105** receives or determines the consumer authentication information, the browser identifier and the sign-on time may vary in different embodiments. For example, in some embodiments the browser identifier may always be determined first and used to determine if the single sign-on feature was previously activated, while in other embodiments a different sequence may be employed. Accordingly, the sequence of exemplary steps **1102 – 1104** is not intended to be limiting.

[0107] At step **1106** the consumer authentication information, the browser identifier, the sign-on time and any other information associated with the sign-on process are stored in an authentication table **113**, which is preferably maintained at the host server **108**. Accordingly, the client-side application **105** may transmit the consumer authentication information, the browser identifier, the sign-on time, etc. to the host server **108**. The host server **108** may utilize the database management system **109** for interacting with the authentication table **113**. The authentication table **113** may alternatively be stored in another location accessible by the host server **108**, such as the data repository **102**, or another network server. Once authenticated, the consumer can access the information account **110** via the vendor web-site **114** using the client device **104**.

[0108] In continuing to operate the browser 112 to access web page files 116 via the network 106, the user may access a subsequent web-site that requires sign-on and authentication to access the consumer information account 110. Like before, upon accessing a new vendor web-site 114, the client device 104 may download an information account-enabled web-page file 116 that is displayed by the browser 112, and the web-page file 116 may include an instruction (e.g., a “call”) that causes the browser 112 to download and execute one or more client-side applications 105. The client-side application 105 responsible for authentication detects a subsequent request for access to the consumer information account 110 via the subsequent web-site at step 1108. As an example, the subsequent request for access to the consumer information account 110 may occur when the user activates an access/create link 908 of the subsequent web-site. When the request for access to the consumer information account 110 is detected, the client-side application 105 determines a browser identifier at step 1110. At step 1112, the browser identifier (as determined at step 1110) may be used to look up the associated consumer authentication information and previous sign-on time stored in the authentication table 113. In particular, the client-side application 105 may transmit the browser identifier (as determined at step 1110) to the database management system 109 at the host server 108, which may access the authentication table 113 to determine the username, password, previous sign-on time, etc. associated with the browser identifier, if any.

[0109] Assuming that consumer authentication information was determined to be associated with the browser identifier, the method next moves to step 1114, where a determination is made as to whether the single sign-on feature is activated. In certain exemplary embodiments, the authentication table 113 may also associate certain preferences with the browser identifier, consumer authentication information, sign-on time, etc. A preference may indicate, for example, whether the user has opted to activate or deactivate the single sign-on feature. By way of example, a dialog box or other interface may be presented to the user during the initial sign-on requesting input from the user as to whether the single sign-on feature should be activated. If single sign-on activation is optional, the database management system 109 (or other responsible network component) may be configured to access the authentication table 113 to determine whether a preference associated with the browser identifier (as determined at step 1110) indicates that the user had previously activated the single sign-on mechanism. Once activated, the single sign-on feature may be automatically

deactivated upon the occurrence of certain terminating events, such as the end of a browser session, a manual sign-off (logout) by the user, the expiration of a time-out interval (see step 1120 below), etc. The user may also be provided with the option to manually deactivate the single sign-on feature.

[0110] If the single sign- feature has not been activated, the method advances to step 1116, whereupon the user is prompted to sign-on again for further access to the information account 110. The user may optionally be prompted with a choice to activate the single sign-on feature during the sign-on process. After the user signs-on via the subsequent web-site, the method returns to step 1104 where the browser identifier and sign-on time are again determined. The method is then repeated from step 1104, as described above.

[0111] On the other hand, if the single sign-on feature has been activated, the method advances from step 1114 to step 1118, whereupon the authentication table 113 is consulted to look up the consumer authentication information and determine if and when the user had been previously authenticated by, for example, determining whether the current browser identifier (as determined at step 1110) matches the most recently stored browser identifier in the authentication table 113. If the browser identifiers do not match, or other specified criteria are not met, the user is considered to not have been previously authenticated and the method proceeds to step 1116 where the user is prompted to sign-on again for further access to the information account 110. If, however, the browser identifiers match, or other specified criteria are met, the consumer is considered to have been previously authenticated and the method advances to step 1120, where it is determined whether an authentication time-out interval has expired.

[0112] An authentication time-out interval may be defined, according to one example, as the maximum permitted duration of time between the occurrence of an event and a subsequent request for access to the information account 110. The event defining the starting point from which the time-out interval will be calculated may be the first manual sign-on, the most previous sign-on (manual or automatic) or other non-sign-on related events. Those skilled in the art will appreciate that the duration of the time-out interval may be specified globally or otherwise by a system administrator or other entity charged with maintaining the data repository 102. When the subsequent request for access to the information account is initiated by the user, the elapsed time between the current time and the occurrence of the starting point event

(e.g., the previous sign-on time) may be determined. If that elapsed time is greater than the duration of the time-out interval, the time-out interval may be considered to have expired. In the preferred embodiment, the time-out interval may be used to enhance the security of the single sign-on mechanism, forcing the user to sign-on again if too much time has elapsed between consecutive sign-on attempts, for example.

[0113] If it is determined at step 1120 that the time-out interval has expired, the method proceeds to step 1116 where the user is prompted to sign-on again for further access to the information account 110. From step 1116 the method returns to step 1104 and is repeated as previously described. However, if it is determined at step 1120 that the time-out interval has not expired, the method advances to step 1122. At step 1122, the vendor server 114 that hosts the subsequent web-site is alerted that the user's previous authentication status remains valid, thus causing the vendor server 114 to by-pass any sign-on interface associated with the information account 110. As an example, the client-side application 105 may receive a message from the host server 108 indicating that the user's previous authentication status remains valid and may pass that message to the vendor server 108 or may generate an instruction that causes the vendor server 108 to by-pass any sign-on interface associated with the information account 110. After an automatic sign-on at step 1122, the method returns to step 1108 to await detection of another request for access to the consumer information account 110 via a subsequent web-site that requires sign-on to access the information account 110.

[0114] Although the single sign-on feature has, in certain instances, been described as being implemented by way of communications between the host server 108 and a client device 104 (e.g., via a client-side application 105), those skilled in the art will appreciate that single sign-on feature may alternately be implemented by way of communications between the host server 108 and a vendor server 114 that hosts a web-site configured to provide access to the central data repository 102 upon authentication of the consumer. Analogously to execution of the client-side applications 105 by the client device 104, the vendor server 114 may execute one or more server-side applications 107 for managing communications with the host server 108 and conducting authentication thereby. Accordingly, one or more server-side applications 107 may be configured to perform the functions of the single sign-on feature, or functions similar thereto, that are described above with respect to one or

more client-side applications 105. In implementing the single sign-on feature through use of server-side applications 107, vendor authentication information and/or an equipment identifier or APPID associated with the vendor server 114 may be transmitted to the host server 108, as appropriate. The vendor server may also communicate with the client device to receive consumer authentication information and/or a browser identifier, if needed.

[0115] As mentioned, once the user is authenticated to access the information account 110, selected consumer information elements may be filtered from the information account 110 and integrated into a vendor's business process on behalf of the user. As an example, the selected consumer information elements may include authentication information (usernames, passwords, biometrics, etc.) that is needed to access secure areas of vendor web-sites. Thus, after the user has successfully signed-on to the information account 110, subsequent authentications of the user for access to the information account 110 may be handled automatically by the single sign-on feature and other consumer authentication information may be auto-populated into sign-on interfaces of secure web-sites on behalf of the consumer. The present invention therefore reduces the consumer's need to repeatedly supply the consumer authentication information for accessing the information account 110 and can virtually eliminate the consumer's need to supply other authentication information for accessing other secure web-sites.

[0116] From a reading of the description above pertaining to various exemplary embodiments, many other modifications, features, embodiments and operating environments of the present invention will become evident to those of skill in the art. The features and aspects of the present invention have been described or depicted by way of example only and are therefore not intended to be interpreted as required or essential elements of the invention. It should be understood, therefore, that the foregoing relates only to certain exemplary embodiments of the invention, and that numerous changes and additions may be made thereto without departing from the spirit and scope of the invention as defined by any appended claims.